Assessing the increase in haptic load when using a dual PHANToM setup

Joan De Boeck, Chris Raymaekers, Karin Coninx

Expertise Centre for Digital Media, Limburg University Centre,
Wetenschapspark 2, B-3590 Diepenbeek, Belgium
{joan.deboeck, chris.raymaekers, karin.coninx}@luc.ac.be

Abstract

Two handed input and collaboration are currently two active research areas in the domain of virtual environments. In a haptic application in particular, by means of a PHANToM device, two possibilities to build such a setup exist: either by integrating two PHANToM interface cards in one computer, or by using two networked computers, each equipped with one PHANToM interface card. The former method is likely to increase the computer's haptic load, while the latter adds extra code complexity and latency in the interaction, but does not augment the haptic load. Since very little can be found about the consequences of both solutions, this paper will describe an experimen, that measures the increase in haptic load of a dual PHANToM setup over a standard configuration.

1 Introduction and related work

Although the use of force-feedback is a huge improvement in the interaction with virtual environments, current implementations are mostly restricted to a single point in space. To achieve a second interaction point, alternatively, a second interface card can be installed in order to drive a second PHANToM device, but this is very likely to increase the computer's haptic load. On the other hand, when using a collaborative application, multiple computers, distributed across the network, are all connected to their own haptic device [ALHAL01][HESPA00]. However, this setup adds extra code complexity and latency in the interaction. As an example, our research into the virtual percussionist application [DEBOE02] adopts those principles of distributed collaborative setups and applies them in a two handed input application. We believe this extra coding complexity might be justified when the increase in the haptic load gets too high in the single computer solution. At this moment however, no formal research has been conducted into this increase in a dual PHANToM setup. Some experiments on the computational load in a single PHANToM setup have been performed: Acosta et al [ACOST02] measured the maximum complexity in terms of number of objects and object complexity across different GHOST versions. Anderson et al [ANDER02] compared the performance of the GHOST API with e-Touch API for large complex objects. This paper, as a part of our research in two-handed haptic input, extents the above-mentioned research. The next sections will describe an experiment, which compares the haptic load of a dual PHANToM setup and a single PHANToM setup and discusses on the results.

2 Experimental Setup

This paper elaborates on two experiments. In a first experiment, we have measured the haptic load in a scene that contains one single object with increasing complexity. This object has been created by subdividing either a tetrahedron or a cube using the Loop subdivision scheme [LOOP87][RAYMA01]. Each subdivision level is represented in the haptic scene graph, by an instance of gstTriPolymeshhaptic. In order to test if the haptic load depends on the object's geometry, we also have used more natural models like a rabbit and a fish.
In a second test, we measured the haptic load in a scene with an increasing number of objects. These objects are positioned in a 3D matrix (as in [ACOST02]) which can grow in each direction. The objects do not intersect each other's bounding box and they also varied in complexity, using the same techniques as in the first experiment.
Both experiments have been conducted with a single and a dual PHANToM setup. The criterium measured in the two experiments is the haptic load, as measured with Sensable's Haptic Load (HLOAD) tool.

The computer used in our experiments was a Pentium III 600 MHz, 256 MB RAM, running Windows NT SP6. However, due to an incompatibility between the PHANToM PCI interface card, the AGP adapter and the computer's motherboard, we were forced to conduct our tests with a poor video card. Since the GHOST thread is a high priority thread, we believe this has little or no effect to the haptic load measured.
Our tests have been conducted on Windows NT, because we did not succeed in connecting two PHANToM devices on Windows 2000[1]. Our assumption that the choice of our hardware and OS does not influence our results are confirmed by comparing our single PHANToM test results on a Pentium III 850 MHz and a dual Pentium III 800 MHz running Windows 2000.

3 Results

3.1 Experiment 1: Objects with increasing complexity

			1 Phantom			2 Phantoms		
Subdivision		**#tri**	**N Contact *1P/0C***	**Contact *1P/1C***	**Contact & Moving**	**N contact *2P/0C***	**1 Contact *2P/1C***	**2 Contacts *2P/2C***
Tetraedron	level 0	4	<10	>10		10	10< - <20	<20
Cube	level 0	12	<10	>10		>10	10< - <20	<20
Tetraedron	level 1	16	<10	>10		>10	10< - <20	<20
Cube	level 1	48	<10	>10		>10	10< - <20	<20
Tetraedron	level 2	64	<10	>10		>10	10< - <20	<20
Cube	level 2	192	<10	>10		>10	10< - <20	20
Tetraedron	level 3	256	<10	>10		>10	10< - <20	20
Cube	level 3	768	<10	>10		>10	10< - <20	>20
Tetraedron	level 4	1024	<10	>10		>10	10< - <20	>20
Cube	level 4	3072	<10	>10		>10	10< - <20	>20
Tetraedron	level 5	4096	<10	>10		>10	10< - <20	>20
Cube	level 5	12288	<10	>10		>10	>20	>30
Tetraedron	level 6	16384	<10	<20		>10	>20	>30
Cube	level 6	49152	<10	20	30	>20	40	<60
Tetraedron	level 7	65536	<10	30	40	<20	40	>70
bunny		69451	<10	50	60	<20	70	quit
Fish		100480	<10	30	90	20	Unstable	quit
rabbit		134074	<10	80	quit	<20	quit	quit
Thetraedron	level8	262144	<10	70	quit	quit	quit	quit

Table 1: haptic load in a single and double PHANToM
Setup with complex objects.

Table 1 summarizes the results of the first experiment. The values indicate a percentage of the haptic load of the simulation. When looking at the first column where no contact is made, one can see a quite constant haptic load both in the single PHANToM (1P/0C) as in the dual PHANToM setup (2P/0C). When touching the object (without moving the surface contact point over the surface) (1P/1C) the haptic load starts increasing from a shape with 160,000 triangles. This is consistent with the values reported in [ANDER01]. As can be seen in the next column, the haptic load augments when the surface contact point moves over the object's surface. This causes the GHOST-thread to quit with the most complex models in our experiment.
With the dual PHANToM condition, the haptic load again is quite stable when no contact is available (2P/0C), but increases slightly with increasing the object's complexity. The haptic load when one PHANToM touches the object (2P/1C) is somewhat higher, compared with the single PHANoM setup. The table here indicates a higher increase for the more complex models. The second surface contact points introduces another increase in such that the total haptic load is roughly 50% more than in (1P/1C). The results of those experiments are graphically depicted in fig 1.

[1] Another research lab reported us the same technical problems when running Windows 2000.

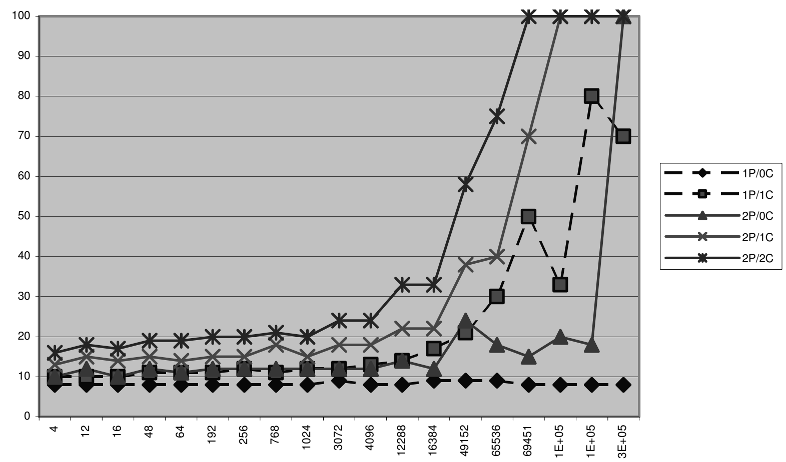

Fig. 1. Graph of the haptic load in a single and double PHANToM Setup with complex objects.

3.2 Experiment 2: Scenes with increasing complexity

Tetrahedron Subdivision Level 0 (4 triangles)					
	One PHANToM		Two PHANToMs		
# objects	**No Contact**	**1 Contact**	**No Contact**	**1 Contact**	**2 Contacts**
8	10	<20	>20	>30	>40
27	<20	>30	30	<50	>60
64	>30	<70	<60	quit	quit
125	<50	quit	quit	quit	quit
216	quit	quit	quit	quit	quit

Table 2: haptic load with multiple objects in the scene. (subdivision level 0)

Tetrahedron Subdivision Level 2 (64 triangles)					
	One PHANToM		Two PHANToMs		
# objects	**No Contact**	**1 Contact**	**No Contact**	**1 Contact**	**2 Contacts**
8	>10	20	<20	>20	30< - <40
27	<20	30< - <40	>30	>50	<70
64	>30	50< - <60	<70	quit	quit
125	<70	quit	quit	quit	quit
216	quit	quit	quit	quit	quit

Table 3: haptic load with multiple objects in the scene. (subdivision level 2)

Tetrahedron Subdivision Level 4 (1024 triangles)					
	One PHANToM		Two PHANToMs		
# objects	**No Contact**	**1 Contact**	**No Contact**	**1 Contact**	**2 Contacts**
8	10	<20	70	quit	quit
27	20	>40	quit	quit	quit
64	<40	60	quit	quit	quit
125	<70	quit	quit	quit	quit
216	quit	quit	quit	quit	quit

Table 4: haptic load with multiple objects in the scene. (subdivision level 4)

Tables 2, 3 and 4 show the results of the haptic load in a scene with an increasing number of objects. Each table displays the same number of objects, but with more triangles per object. All the objects are placed in a grid in such a manner that their bounding boxes do not overlap. A single PHANToM setup can fully support a scene with up to 64 tetrahedrons, while the dual PHANToM setup has the same load when simulating only 27 objects. The results of table 4 show that the haptic loop with 2 PHANToMs quits when touching one of the level 4 subdivision-objects in the scene, while the single PHANToM setup still supports 64 objects.

4 Discussion

4.1 Results

Our values in the single PHANToM case correspond to the findings of [ACOST02] and [ANDERS02]. Although, [ACOST02] can support up to 600 cubes, while our simulation already quits at 125 objects. We suppose this is caused by the standard *gstCube* (used in [ACOST02]), which is simpler and more efficient than the *gstTriPolymesh* used in our experiment.

From the first experiment, we can conclude that the haptic load in a dual PHANToM setup has been increased, compared to the single PHANToM setup. As long as the object is relatively simple (up to 16,000 triangles) the haptic load keeps below 30%, which results in a stable simulation.

Although the HLOAD application, which was the only tool we had to measure the haptic load, is not the most accurate tool one can imagine, we roughly can say that the second PHANToM increases the haptic load with about 50%. This can be confirmed by comparing the number of triangles in the most complex, but stable simulation in the first condition (fish with 100,480 triangles) with the maximum number of triangles in a stable dual PHANToM simulation (tetrahedron level 7 with 65,536 triangles).

Even more pronounced is the increase of the haptic load in a scene with multiple objects. If we compare “single contact with one PHANToM” (1P/1C) with a “double contact with two PHANToMs” (2P/2C), we see that the haptic load almost doubles. This makes that complex scenes, which can be run with one PHANToM, are not supported by a dual PHANToM setup.

4.2 Other findings

During the course of our experiments, we have encountered a number of interesting situations. Some of these appear to be obvious, but we believe they can give the programmer a better understanding of optimizing a more complex scene.

- When starting the haptic loop, very often a “haptic load spike” is encountered. This is why heavy models often crash at the beginning of the simulation. However if a complex simulation accidentally “survives” a startup, the simulation seldom is completely stable. Most of the time the haptic thread quits when acting on the scene.
- When exploring our “natural” objects (fish, rabbit), the haptic load was higher when approaching a more complex region with lots of small triangles.
- When sliding the PHANToM over an object, this increases the haptic load, compared to a static contact. On the other hand a static contact, which touches more than one triangle (e.g. in a corner), requires more processing time.
- We could not make Windows 2000 to work with the two PHANToMs, although we have conducted some of our single PHANToM tests on two Windows 2000 PCs, as well. In general we can state that there is little difference between the results of these tests and the Windows NT test.
- At first sight, the dual processor seems to perform better in starting-up a very complex scene: scenes that quit at start-up, do run on the dual processor computer. When interacting in those scenes, however, the results are quite similar to a single processor machine.
- On a dual processor computer other tasks will slow down less when executing a heavy GHOST thread. For instance, when running complex scenes, the haptic loop will slow down the graphics on a single processor computer, which is not true on a dual processor machine. This is quite obvious because the ghost thread in some cases can take up to about 99% of one processor’s time, which is only 50% of the total processing power of a dual machine.

5 Conclusions and future work

For a multiple-contact interaction with a PHANToM device, two possibilities exist: most commonly, a second PHANToM will be attached to the same computer, or alternatively two computers in a network, both connected to a separate PHANToM share the same virtual scene. The first solution increases the computational load; the next solution introduces network delays. As a step to a well-grounded choice between those two options, in this paper we have conducted a test to measure the increase of the haptic load of a common dual PHANToM setup. Because we could not make a dual setup to work under Windows 2000, we have conducted our experiment under Windows NT SP6.

The experiment consisted of two tests: one that measured the haptic load in respect to the complexity of one object, and a second experiment, which tested the haptic load in respect to the scene complexity.
We can conclude that the increase of the haptic load is quite significant (roughly about 50% in experiment 1 and about 100% in experiment 2), but this is of less importance when running small scenes (1 object of less than 16,000 triangles, or a scene of less than 30 objects). We want to conclude that for large or complex scenes the dual PHANToM setup clearly has its limitations. In such a case, one can consider to afford the extra code complexity for a distributed setup, although network delay certainly will be a constraint in this case.

6 Acknowledgements

We first of all want to thank Geert Van De Boer for his coding effort in realizing this test and getting a dual PHANToM setup to work. Next, we also want to thank Jan van Erp en Hendrik-Jan van Veen from TNO (the Netherlands) for their suggestion to install Windows NT instead of Windows 2000.

7 References

[ACOST02] Scene Complexity: A measure for real-time stable haptic applications, E.Acosta, B.Temkin, October 2001, Aspen; CO, USA. 2001

[ALHAL01] Tele-Handshake: A Cooperative Shared Haptic Virtual Environment, M.O.Alhalabi, S.Horiguchi, EuroHaptics 2001, July 2001, Birmingham, UK.

[ANDER02] The ActivePolygon Polygonal Algorithm for Haptic Force Generation, T. Anderson, N. Brown, Novint Technologies, PUG2001, October 2001, Aspen; CO, USA. 2001

[DEBOE02] A networked two-handed haptic experience: The Virtual Percussionist, J. De Boeck, P. Vandoren, K. Coninx, VRIC2001, June 2002, Laval, France, pp 131-139

[HESPA00] Haptic Collaboration over the Internet, J. Hespana, M. McLaughlin, G. Sukhatme, M. Akbarian, R. Grag, W. Zhu, Proceedings of the fifth PHANToM Users Group Workshop; October 28-30, Aspen; CO, USA. 2000

[LOOP87] Smooth Subdivision Surfaces Based on Triangles, C. Loop, University of Utah, Department of Mathematics, Utah, US, August 1987

[RAYMA01] Fast Haptic Rendering of Complex Objects Using Subdivision Surfaces, C. Raymaekers, K. Beets; F. Van Reeth, PUG2001, October 2001, Aspen; CO, USA. 2001